

# PENENTUAN FASE LALU LINTAS PADA PERSIMPANGAN DENGAN MENGGUNAKAN PENDEKATAN TEORI GRAF

I Made Mertha Ariada<sup>(1)</sup>, R. Gunawan Santosa<sup>(2)</sup>, Nugroho Agus Haryono<sup>(3)</sup>

## Abstrak:

Masalah yang dihadapi dalam Tugas Akhir ini adalah bagaimana mengatur fase lalu lintas supaya tidak terjadi tabrakan. Permasalahan ini akan diselesaikan dengan menggunakan pendekatan teori graf. Untuk melakukan pemecahan terhadap masalah diatas adalah dengan membangun aplikasi pengaturan fase lalu lintas dengan menggunakan pendekatan teori graf dengan metode graf kesesuaian. Untuk menangani masalah grafis dilakukan perhitungan sudut dengan menggunakan rumus trigonometri, disamping itu juga menggunakan rumus untuk mencari persamaan garis dan titik potong pada dua garis.

**Kata Kunci:** *Lalu lintas, Persimpangan, Graf, Compatibility Graph, Graf Kesesuaian.*

## 1. Pendahuluan

Semakin lama, kebutuhan manusia untuk melakukan aktifitas diluar rumah semakin basar. Untuk menuju tempat tujuan, orang sering menggunakan alat transportasi seperti sepeda, sepeda motor ataupun mobil. Jumlah dari kendaraan transportasi juga semakin banyak. Kemungkinan besar suatu jalan akan bertemu dengan jalan lain disuatu lokasi. Lokasi pertemuan dua jalan atau lebih sering disebut dengan simpang. Untuk menghindari akibat yang berbahaya seperti tabrakan pada persimpangan jalan, dibuat lampu lalu lintas dengan tiga warna yaitu merah, kuning, dan hijau.

Permasalahannya adalah bagaimana mengatur fase lalu lintas agar tidak terjadi tabrakan. Permasalahan ini akan diselesaikan dengan menggunakan pendekatan teori graf.

## 2. Landasan Teori

### a. Graf

Graf adalah diagram yang memuat titik-titik yang disebut verteks yang masing-masing dapat dihubungkan oleh garis yang biasa disebut edge. Graf biasanya disajikan dalam bentuk gambar, tetapi agar sistem dapat menerima objek sebagai graf maka graf tersebut diterjemahkan dalam bentuk matrik terlebih dahulu. Matrik yang digunakan adalah matriks *adjacency* yaitu matriks berukuran  $n \times n$  dengan setiap elemen baris  $i$  kolom  $j$  adalah jumlah edge yang menghubungkan verteks  $i$  dengan verteks  $j$ . *Adjacency list* adalah suatu daftar dari matrik *adjacency* yang menunjukkan bahwa suatu verteks berhubungan dengan verteks mana saja. Hubungan verteks ini digambarkan dalam bentuk *link list*.

Dalam graf terdapat banyak jenis operasi, salah satunya adalah operasi *Fusi*

<sup>(1)</sup> I Made Mertha Ariada, Mahasiswa Teknik Informatika, Fakultas Teknik, Universitas Kristen Duta Wacana

<sup>(2)</sup> Drs. R. Gunawan Santosa, M. Si, Dosen Teknik Informatika, Fakultas Teknik, Universitas Kristen Duta Wacana

<sup>(3)</sup> Nugroho Agus Haryono, S.Si., M.Si., Dosen Teknik Informatika, Fakultas Teknik, Universitas Kristen Duta Wacana



(penggabungan). Sepasang verteks  $v_1$  dan  $v_2$  pada graf  $G$  dikatakan digabung atau difusikan bila kedua verteks tersebut diganti oleh verteks baru yang tunggal yaitu  $v_F$  sedemikian sehingga setiap edge yang berpengaruh pada  $v_1$  atau  $v_2$  atau kedua-duanya berpengaruh pada verteks baru tersebut. Dalam *fusi* atau penggabungan, jumlah edge dari verteks  $v_3$  ke  $v_F$  adalah jumlah edge dari  $v_1$  ke  $v_3$  ditambah dengan jumlah edge dari  $v_2$  ke  $v_3$ .

#### **b. Graf Kesesuaian**

Graf kesesuaian sering digunakan dalam memecahkan masalah dengan cara menyusun data dalam urutan fakta-fakta. Pada graf kesesuaian, verteks mewakili objek yang tersusun, dan edge menghubungkan pasangan verteks yang sesuai dengan syarat yang diinginkan. Dalam kasus ini, graf kesesuaian digunakan untuk mengubah diagram dari arus lalu lintas kedalam bentuk graf dimana verteks mewakili aliran trafik dan edge mewakili pasangan verteks yang sesuai (tidak menimbulkan bahaya tabrakan).

#### **c. Koordinat**

Sumbu simetri koordinat digambarkan dengan dua garis riil, satu mendatar dan yang lainnya tegak sedemikian sehingga keduanya berpotongan pada titik nol dari kedua garis tersebut. Dua garis itu dinamakan sumbu-sumbu koordinat, perpotongannya diberi label  $O$  dan disebut titik asal. Garis yang mendatar disebut sumbu  $x$  dan garis yang tegak disebut sumbu  $y$ .

Lingkaran yang dibentuk dengan pusat titik awal dan jari-jari yang ditentukan, sehingga lingkaran itu dapat dibagi beberapa bagian yang simetris dengan luas yang sama dengan membagi dari titik awal yang sering disebut titik pusat lingkaran.

#### **d. Kemiringan dan Perpotongan Garis**

Kemiringan adalah suatu angka yang dapat bernilai positif atau negatif yang merepresentasikan nilai tegak dari suatu garis atau dua titik. Nilai kemiringan nantinya dapat dimanfaatkan untuk mencari rumus fungsi garis lurus yang dibentuk dari titik-titik yang sama yang digunakan untuk mencari kemiringan. Perpotongan garis didapat dengan menggambar dua garis, jika pada suatu titik kedua garis itu bertemu maka titik itu adalah titik perpotongan garis.

### **3. Pembahasan**

#### **a. Algoritma**

Algoritma untuk menentukan fase lalu lintas yang melewati persimpangan adalah :

- 1) Mulai
- 2) *Input* berupa simpang yang akan diteliti.
- 3) *Input* berupa arah dari setiap jalur dan nama jalan pada persimpangan.
- 4) Buat sebuah diagram dengan jumlah titik sama dengan dua kali jumlah simpang dan dibuat melingkar sesuai dengan urutan jalur kendaraan.
- 5) Dalam diagram dibuat garis menghubungkan setiap titik dari jalur menuju persimpangan dengan semua jalur yang mengarah keluar persimpangan, kecuali pada jalan yang sama.
- 6) Pilih satu garis sebagai kunci, dan tentukan garis lain yang tidak berpotongan dengan garis kunci didalam lingkaran, simpan dalam adjacency list.
- 7) Lakukan langkah 6 sampai semua garis dipilih sebagai kunci.
- 8) Dari adjacency list tersebut, buat sebuah graf kesesuaian dimana setiap verteks

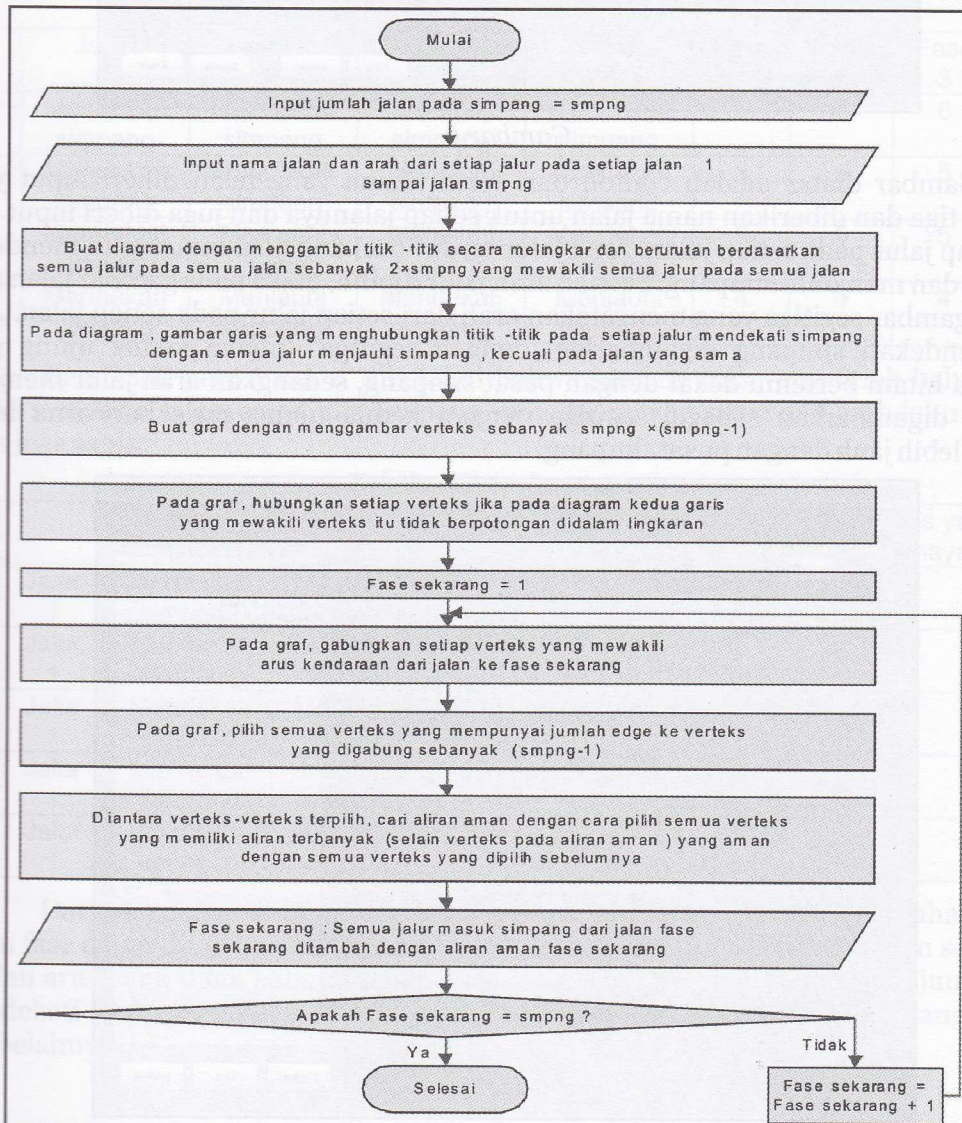


mewakili garis kunci dan edge mewakili garis antara verteks kunci dengan semua verteks dalam adjacency listnya.

- 9) Buat graf baru dengan memilih semua verteks yang mewakili kunci dari arah jalan yang sama kemudian digabungkan, lalu pilih semua verteks yang dihubungkan oleh edge sebanyak simpang 1 dengan verteks yang digabung dan pilih juga aliran lain yang tidak menimbulkan bahaya tabrakan dengan cara melihat apakah semua verteks yang dipilih dihubungkan oleh edge.
- 10) Lakukan langkah 9 sampai setiap verteks yang berasal dari arah jalan yang sama telah digabungkan (jumlah graf baru yang dibuat sama dengan jumlah jalan pada simpang).
- 11) Verteks-verteks yang terpilih dari setiap graf baru 1 sampai n mewakili aliran trafik yang dapat melewati persimpangan dengan tidak menimbulkan bahaya tabrakan.
- 12) Selesai.

## b. Flowchart

Flowchart secara umum untuk menentukan fase lalu lintas yang dapat melewati persimpangan:



Gambar 1



### C. Implementasi

**Form Utama**

Menu

**Penentuan Fase Lalu-lintas Pada Persimpangan Dengan Menggunakan Pendekatan Teori Graf**

Pilih simpang yang akan diteliti

☐ Simpatig Tiga  
☐ Simpatig Empat  
☐ Simpatig Lima  
☐ Simpatig Enam

Isi nama jalan dan arah jalur kendaraan

Nama jalan 1 adalah:   
 Arah jalur 1 adalah:   
 Arah jalur 2 adalah:   
 Nama jalan 2 adalah:   
 Arah jalur 3 adalah:   
 Arah jalur 4 adalah:   
 Nama jalan 3 adalah:   
 Arah jalur 5 adalah:   
 Arah jalur 6 adalah:

\* Nama jalan maksimal 30 karakter  
 \* Untuk mengaktifkan tombol Simulasi, semua arah jalur dan nama jalan harus terisi dengan benar

Buka Simpan Simulasi Keluar

Gambar 2

Gambar diatas adalah contoh dari form utama yang telah diberi input yaitu simpang tiga dan diberikan nama jalan untuk setiap jalannya dan juga diberi input arah dari setiap jalur pada setiap jalan. Arah jalur terdiri dari dua nilai input yaitu mendekati simpang dan menjauhi simpang. Setelah arah jalur dipilih, maka pada gambar jalan akan muncul gambar segitiga yang menyatakan arah dari setiap jalur pada setiap jalan. Arah jalur mendekati simpang digambarkan dengan segitiga dimana kedua ujung garis berwarna hitam bertemu dekat dengan pusat simpang, sedangkan arah jalur menjauhi simpang digambarkan sebagai segitiga dengan kedua ujung garis berwarna hitam bertemu lebih jauh dengan pusat simpang.

**Simulasi**

**Simulasi Fase Lalu-lintas**

☐ Fase ke 1  
☐ Semua Per Fase  
☐ Semua Fase

Keterangan arus kendaraan dari setiap fase

Fase ke 1  
 Jl. Mawar ke Jl. Anggrek  
 Jl. Mawar ke Jl. Melati  
 Jl. Anggrek ke Jl. Mawar

Diagram Graf Kembali

Gambar 3



Output dari aplikasi ini berupa tiga jenis simulasi yaitu simulasi “fase ke” yang menampilkan gambar simpang pada fase sesuai nilai dalam numeric up down, simulasi “Semua Per Fase” yang menampilkan semua fase secara bergantian dengan jeda waktu tertentu dimulai dari fase pertama sampai fase ke n dengan n adalah jumlah jalan pada simpang., dan simulasi “Semua Fase” yang menampilkan semua fase secara bergantian dengan jeda waktu tertentu dimulai dari fase pertama sampai fase ke n dengan n adalah jumlah jalan pada simpang.. Dari contoh input, akan menghasilkan output seperti gambar diatas. Form simulasi secara default menampilkan radio button “fase ke” dalam keadaan terpilih dengan nilai 1 pada numeric up down-nya sehingga akan menampilkan gambar simpang tiga dengan fase ke 1. Untuk mempermudah dalam memahami arti dari garis arus kendaraan yang dibuat maka dibuat sebuah textbox yang berisi keterangan dari fase yang sedang aktif dan keterangan arah arus kendaraan dari jalan asal ke jalan tujuan.

#### 4. Analisis

Dalam analisis diambil contoh pada simpang empat dengan melakukan dua buah percobaan.

Tabel 1 : Percobaan 1

	Jalan 1	Jalan 2	Jalan 3	Jalan 4	Fase 1	Fase 2	Fase 3	Fase 4
Jalur 1	Mendekati simpang	Mendekati simpang	Mendekati simpang	Mendekati simpang	6	6	6	6
Jalur 1	Mendekati simpang	Mendekati simpang	Menjauhi simpang	Mendekati simpang	5	5	5	5
Jalur 1	Mendekati simpang	Menjauhi simpang	Menjauhi simpang	Mendekati simpang	5	5	5	5
Jalur 1	Mendekati simpang	Menjauhi simpang	Mendekati simpang	Menjauhi simpang	4	4	4	4

Dari hasil percobaan 1, diketahui bahwa jumlah arus yang tidak bahaya pada setiap fase akan selalu sama jika simpang yang digunakan sama dan arah jalur dari tiap jalan juga sama.

Tabel 2 : Percobaan 2

	Jalan 1	Jalan 2	Jalan 3	Jalan 4	Jumlah arus yang tidak bahaya
Jalur 1	Mendekati simpang	Mendekati simpang	Mendekati simpang	Mendekati simpang	6
Jalur 1	Mendekati simpang	Mendekati simpang	Mendekati simpang	Menjauhi simpang	5
Jalur 1	Mendekati simpang	Mendekati simpang	Menjauhi simpang	Menjauhi simpang	5
Jalur 1	Mendekati simpang	Menjauhi simpang	Mendekati simpang	Menjauhi simpang	4
Jalur 1	Menjauhi simpang	Menjauhi simpang	Menjauhi simpang	Menjauhi simpang	6

Dari percobaan 2 dapat diketahui bahwa jumlah arus yang tidak bahaya dalam suatu fase dapat dimaksimalkan dengan membuat arah jalur yang sama dan sebaliknya jumlah arus yang tidak bahaya dalam suatu fase akan menjadi sedikit jika jumlah arah mendekati dan menjauhi sebanding dan berselang seling dengan jalan yang terletak disebelahnya.



## 5. Kesimpulan

Setelah program aplikasi selesai dan dilakukan beberapa percobaan, didapat beberapa kesimpulan antara lain:

- a. Aplikasi ini dapat menentukan fase lalu lintas pada persimpangan dengan meminimalkan bahaya kecelakaan.
- b. Aplikasi ini dapat diterapkan pada persimpangan yang sesuai dengan batasan masalah yang diberikan.
- c. Kelebihan dari aplikasi ini adalah mampu melakukan penentuan fase lalu lintas dengan cepat jika dibandingkan dengan penentuan fase lalu lintas secara manual.
- d. Jumlah arus yang tidak bahaya pada setiap fase akan selalu sama jika simpang yang digunakan sama dan arah jalur dari tiap jalan juga sama.
- e. Jumlah arus yang tidak bahaya dalam suatu fase dapat dimaksimalkan dengan membuat semua arah jalur pada jalur ganjil mengarah ke arah yang sama, misalnya mendekati simpang. Jumlah arus yang tidak bahaya dalam suatu fase akan menjadi minimal jika jumlah arah mendekati dan menjauhi simpang sebanding dan berselang seling dengan jalan yang terletak disebelahnya.
- f. Jumlah maksimal arus yang tidak bahaya dalam suatu fase adalah  $2 \times (\text{jumlah jalan pada simpang} - 1)$

## 6. Daftar Pustaka

- Goldstein, Larry J., David C. Lay and David I. Schneider (1984), Calculus and Its Applications, third edition, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- Jaenudin, ST (2006), belajar sendiri .net dengan visual C# 2005, Edisi 1, Penerbit Andi, Yogyakarta.
- Pangajow, Frank Albert (1996), Serial Grafik dari Frank, jilid 1, penerbit Dinastindo, Jakarta.
- Purcell, Edwin J. dan Dale Varberg, Kalkulus dan Geometri Analitis, jilid 1. Terj. Drs. I Nyoman Susila, M. Sc, Penerbit Erlangga.
- Troelsen, Andrew (2001), C# and the .Net Platform, Springer-Verlag GmbH & Co. KG, Germany.
- West, Douglas Brent (2001), Introduction To Graph Theory, second edition, Prentice-Hall, Inc. Upper Saddle River, NJ.
- Wilson, Robin J. dan John J. Watkins (1990), Graphs An Introductory Approach, A First Course in Discrete Mathematics, John Wiley & Sons, Canada.